

What is claimed is:

1 1. An apparatus comprising:

2 an execution unit to execute an instruction;

3 a replay system to replay an altered instruction if the execution unit executes

4 the instruction erroneously.

1 2. The apparatus of claim 1 wherein the replay system comprises:

2 a replay loop to replay the instruction under a first condition; and

3 an instruction morphing circuit to replay the altered instruction under a second
4 condition.

1 3. The apparatus of claim 1 wherein the replay system comprises:

2 a replay loop to replay the instruction if the instruction is a first instruction;

3 and

4 an instruction morphing circuit to replay the altered instruction if the
5 instruction is a second instruction.

1 4. The apparatus of claim 3 wherein the first instruction is one of a plurality of non-

2 modifiable instructions and the second instruction is one of a plurality of modifiable
3 instructions.

1 5. The apparatus of claim 4 wherein the plurality of modifiable instructions are morphed

2 only if a failure in their initial execution occurs.

1 6. The apparatus of claim 1 wherein said replay system tracks at least one extra bit to
2 allow alterations of instructions.

1 7. The apparatus of claim 1 wherein said apparatus comprises a low level cache and a
2 higher level cache, wherein the replay system is to alter a load instruction that has
3 already missed in the higher level cache to thereafter only access the low level cache.

1 8. The apparatus of claim 1 wherein said apparatus comprises a page miss handler to
2 handle instructions that cause page faults, wherein the instruction is a memory access
3 that causes a page fault, and wherein the replay system is to change the memory
4 access to one or more memory accesses to handle the page fault.

1 9. The apparatus of claim 8 wherein the replay system is to replace the memory access
2 with a page descriptor read, then to replace said page descriptor read with a page table
3 entry read, then to reinstate the memory access.

1 10. The apparatus of claim 1 wherein said instruction is a dependent instruction that is
2 dependent on a result from a previous instruction, and wherein the replay system is to
3 alter the dependent instruction to avoid execution in further iterations through the
4 replay system until the previous instruction has successfully executed.

1 11. The apparatus of claim 10 wherein the replay system is to alter the dependent
2 instruction by setting a valid bit for the dependent instruction to indicate that the

3 instruction is invalid.

1 12. The apparatus of claim 10 wherein the replay system is to alter the dependent
2 instruction back into an executable form when said previous instruction retires.

1 13. The apparatus of claim 11 wherein the replay system is to reset the valid bit when any
2 instruction retires.

1 14. The apparatus of claim 10 wherein the replay system is to track a sequence number
2 for the previous instruction and wherein the replay system is to return the dependent
3 instruction to an executable form when said previous instruction completes.

1 15. The apparatus of claim 10 wherein the apparatus further includes a cache, and
2 wherein the replay system is to return the dependent instruction to an executable form
3 when a write to the cache occurs.

1 16. The apparatus of claim 1 wherein said instruction is a high precision instruction and
2 said replay system is to generate a first result and then the altered instruction is to be
3 executed to generate a final result from the first result.

1 17. The apparatus of claim 1 wherein said execution unit is a numeric execution unit and
2 wherein said replay system is to detect a data dependent condition for the instruction
3 and to provide the altered instruction to achieve an identical result.

1 18. The apparatus of claim 17 wherein the instruction is a rounding instruction and the
2 altered instruction is an add instruction.

1 19. The apparatus of claim 17 wherein the numeric execution unit lacks hardware to
2 compute one or more relatively rare numeric cases and wherein such relatively rare
3 numeric cases are instead implemented by injecting, via the replay system, the altered
4 instruction to achieve an effectively identical result.

1 20. A processor comprising:
2 a scheduler to dispatch an original instruction;
3 an execution unit to attempt execution of the original instruction;
4 a checker to determine whether the original instruction executed properly;
5 a replay system comprising:
6 a replay loop to replay the original instruction;
7 a morphing circuit to change the original instruction into an altered
8 instruction and to replay the altered instruction.

1 21. The processor of claim 20 wherein said replay system is coupled to replay the original
2 instruction when a first condition occurs and to replay the altered instruction when a
3 second condition occurs.

1 22. The processor of claim 20 wherein said replay system is coupled to replay the original
2 instruction when the original instruction is a first instruction and to replay the altered

3 instruction when the original instruction is a second instruction.

1 23. The processor of claim 22 wherein said first instruction is one of a plurality of non-
2 alterable instructions and wherein said second instruction is one of a plurality of
3 alterable instructions.

1 24. A method comprising:
2 executing an original instruction;
3 determining if a first condition occurs;
4 if said first condition occurs, then
5 morphing said original instruction to form a morphed instruction; and
6 executing said morphed instruction.

1 25. The method of claim 24 wherein determining if the first condition occurs further
2 comprises:
3 determining whether the original instruction executed improperly.

1 26. The method of claim 24 further comprising:
2 determining whether a second condition occurs;
3 if said second condition occurs, then
4 replaying said original instruction for execution.

1 27. The method of claim 24 wherein morphing comprises:

2 altering a load instruction that has already missed in a higher level cache to
3 thereafter only access a lower level cache.

1 28. The method of claim 24 wherein morphing comprises:

2 altering a page-fault-causing instruction to perform one or more other
3 instructions to handle a page fault.

1 29. An article comprising a machine readable medium that stores data representing an
2 integrated circuit comprising:

3 an execution unit to execute an instruction;
4 a replay system to replay an altered instruction if the execution unit executes
5 the instruction erroneously.

1 30. The article of claim 29 storing further data representing the integrated circuit, which
2 further comprises:

3 a replay loop to replay the instruction under a first condition; and
4 an instruction morphing circuit to replay the altered instruction under a second
5 condition.

1 31. The article of claim 29 wherein the data representing the integrated circuit comprises
2 a functional description of the integrated circuit.

1 32. The article of claim 29 wherein the data representing the integrated circuit comprises

2 a hardware description language code.

1 33. The article of claim 29 wherein the data representing the integrated circuit comprises
2 data representing a plurality of mask layers string physical data representing the
3 presence or absence of material at various locations of each of said plurality of mask
4 layers.

1 34. An article comprising a machine readable carrier medium having stored thereon data
2 which, when loaded into a computer system memory in conjunction with simulation
3 routines, provides functionality of a model comprising:

4 an execution unit to execute an instruction;

5 a replay system to replay an altered instruction if the execution unit executes
6 the instruction erroneously.

1 35. The article of claim 34 wherein the model further comprises:

2 a replay loop to replay the instruction under a first condition; and

3 an instruction morphing circuit to replay the altered instruction under a second
4 condition.